Oracle® Server Management Agents User's Guide

Part No: E52098-08
January 2017

Oracle Server Management Agents User's Guide

Part No: E52098-08

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Référence: E52098-08

Accessibilité de la documentation

Pour plus d'informations sur l'engagement d'Oracle pour l'accessibilité à la documentation, visitez le site Web Oracle Accessibility Program, à l'adresse `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

Accès aux services de support Oracle

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` ou le site `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` si vous êtes malentendant.

Contents

Using This Documentation

This section describes product information, documentation and feedback, and a document change history.

Documentation and Feedback

The following documentation is available related to the Oracle Hardware Management Pack.

Documentation	Link
All Oracle products	https://docs.oracle.com
Oracle Hardware Management Pack	http://www.oracle.com/goto/ohmp/docs
Oracle ILOM	http://www.oracle.com/goto/ilom/docs

Provide feedback on this documentation at:

http://www.oracle.com/goto/docfeedback

Support and Training

These web sites provide additional resources:

- Support: https://support.oracle.com
- Training: http://education.oracle.com

Contributing Authors

The following authors contributed to this documentation: Cynthia Chin-Lee, Lisa Kuder, David Moss, Ralph Woodley, Michael Bechler.

Change History

The following changes have been made to the documentation set.

- May 2014. Initial publication.
- June 2014. Updated *Release Notes* to add issue 18866436. Updated the Hardware Management Agent overview description in the *Installation Guide* and *Management Agents User's Guide*. Updated the *CLI User's Guide* to add a procedure on checking the Host-to-ILOM Interconnect connection.
- August 2014. Added a note regarding Oracle Solaris 11.2 integration. Updated the *CLI User's Guide* to make editorial improvements. Updated the *Installation Guide* to document an installation issue and updated the *Management Agents User's Guide* to make minor technical updates.
- June 2015. Updated the *Release Notes* to include Oracle Hardware Management Pack 2.3.2.2 updates. Updated the *CLI User's Guide* to add error codes for the `ubiosconfig` command and added information on updating firmware on QLogic and Emulex fiber channel controllers. Updated the *Installation Guide, Management Agents User's Guide* and *CLI User's Guide* to make editorial improvements and other minor technical updates.
- July 2015. Updated *Release Notes* for minor editorial change. Updated the *Management Agents User's Guide* add additional information on Sun Storage 6 Gb SAS PCIe HBA disk events. Other minor editorlal updates made to library.
- September 2015. Updated *Release Notes* to include Oracle Hardware Management Pack 2.3.3 updates. Updated the *Installation Guide* and *Linux FMA User's Guide* to add information on disabling EDAC. Updated *CLI User's Guide* to describe manual power cycle requirement for remote service processor firmware updates. Updated *Management Agents User's Guide* snmpwalk and set examples. Other minor editorial updates made to library.
- May 2016. Updated the *Release Notes*, *Installation Guide* and *Management Agents User's Guide* for bug 23299302.
- January 2017. Updated the *Release Notes* to include Oracle Hardware Management Pack 2.3.8 updates. Updated the *License Information User Manual* to provide attributions for included third-party software. Minor editorial updates to the *Management Agents User's Guide*.

Oracle Server Management Agents User's Guide Overview

This guide provides an overview of Oracle Server Management Agents (Management Agents) and how to use them with your Oracle server.

Note - This documentation applies to servers running Oracle Solaris 11.1 and earlier **or** other supported operating systems.

Beginning with Oracle Solaris 11.2, Oracle Hardware Management Pack (HMP) has become an integrated component of the operating system and is called Oracle HMP for Oracle Solaris. Do **not** download and use other versions of Oracle Hardware Management Pack that are not specifically qualified for the Oracle Solaris 11.2 (and later) operating system. For details, see `http://www.oracle.com/goto/ohmp/solarisdocs`. If you have Oracle Solaris 11.1 or earlier or other operating systems, continue to use Oracle HMP. It is available as a separate download from `http://support.oracle.com`.

This guide includes:

For information on installing Management Agents, see *Oracle Hardware Management Pack Installation Guide*.

Oracle Server Management Agents

Oracle Server Management Agents provide operating system-specific agents to manage and configure your Oracle servers.

Oracle Server Management Agents include:

- Oracle Hardware Management Agent – Runs in the background to collect information about the system to support Oracle ILOM and SNMP monitoring.
- Oracle Server SNMP Plugins - System Management Information Base (MIBs) that support native SNMP monitoring.
- Support for SNMP trap forwarding using a utility called `itpconfig` to send SNMP traps between Oracle ILOM and an SNMP trap destination over the Host-to-ILOM Interconnect
- Oracle Hardware Management Pack Watchdog Agent – Used on Linux systems to alert you if Oracle ILOM or the host become unresponsive and to preform a corrective action.

For a description of these components, see:

- "Oracle Server Hardware Management Agent" on page 13
- "Oracle Server Hardware SNMP Plugins" on page 14
- "itpconfig and the ILOM Trap Proxy" on page 15
- "Oracle Hardware Management Pack Watchdog Agent" on page 15

Oracle Server Hardware Management Agent

The Oracle Server Hardware Management Agent (Hardware Management Agent) and associated Oracle Server Hardware SNMP Plugins (Hardware SNMP Plugins) provide a way to monitor and manage your server and server module's hardware using an operating system native agent.

This in-band functionality enables you to use a single IP address (the host's IP) for monitoring your servers and blade server modules, without having to connect the management port of the Oracle Integrated Lights Out Manager (ILOM) service processor to the network.

The Hardware Management Agent and Hardware SNMP Plugins run on the host operating system of your Oracle servers, communicating with the Oracle ILOM service processor. The Hardware Management Agent daemon, called hwmgmtd, regularly polls the service processor for information about the current state of the server. Hardware Management Agent can poll the service processor for hardware information over either the Host-to-ILOM Interconnect, available on Oracle latest servers, or KCS interface on previous generation servers. This information is then made available by Hardware Management Agent over SNMP using the Hardware SNMP Plugins.

In addition, the Hardware Management Agent maintains a separate log that contains information about the Hardware Management Agent status, which can be used for troubleshooting.

Note - In previous versions of Hardware Management Pack (before version 2.3.0.0), the agent read the service processor's System Event Log (SEL) records, logged new events to syslog and generated SNMP traps using the host SNMP daemon. As of version 2.3.0.0, the Hardware Management Agent no longer performs this function.

To replicate this functionality, use the `ipmievd` daemon (available as part of ipmitool with Oracle Hardware Management Pack) to log SEL events to syslog. The Oracle ILOM Trap proxy (`itpconfig`) can also be used to forward Oracle ILOM generated SNMP traps using the Host-to-ILOM interconnect.

Oracle Server Hardware SNMP Plugins

The Oracle Server Hardware SNMP Plugins consists of Net-SNMP plugins, that are versions of hardware-specific Management Information Bases (MIB) which have been designed to enable you to monitor your Oracle servers effectively.

The sunHwMonMIB describes the state of sensors and alarms on your servers and provides the following information:

- Overall system alarm status
- Aggregate alarm status by device type
- FRU Alarm status
- Lists of sensors, sensor types, sensor readings, and sensor thresholds
- Indicator states
- System locator control
- Inventory including basic manufacturing information

- Product and chassis inventory information (such as serial number and part numbers)
- Per-sensor alarm status

The sunHwTrapMIB describes a set of traps for hardware events that can be generated by an Oracle server and provides the following information:

- Conditions affecting the environmental state of the server (such as temperature, voltage, and current out-of-range conditions)
- Error conditions affecting the hardware components in the server such as FRU insertion and removal and security intrusion notification

The sunStorageMIB provides the following information about system storage:

- Basic manufacturing information, properties, and alarm status for controllers
- Properties and alarm status for disks
- Properties and alarm status for RAID volumes
- Status of logical components

itpconfig and the ILOM Trap Proxy

The `itpconfig` command line tool allows you to configure a trap forwarding proxy to forward Oracle ILOM generated SNMP traps to the host or a configured SNMP trap destination over the Host-to-ILOM Interconnect. This allows SNMP traps to be forwarded to a destination you specify without having to have a network connection to the server's NET MGT port.

The `itpconfig` command can be used to both setup a trap proxy and to configure the Host-to-ILOM Interconnect between the Oracle ILOM service processor and the host.

Refer to your server documentation to see if your server supports the Host-to-ILOM Interconnect.

Oracle Hardware Management Pack Watchdog Agent

The Oracle Hardware Management Pack Watchdog Agent (HMP Watchdog Agent) is an optional component that works with Linux. The HMP Watchdog Agent periodically checks the host and/or Oracle ILOM and performs a user-configured action if either proves unresponsive. The actions can include posting a warning to a log file, resetting the corresponding device, and in the case of the host, power cycling or powering off the host.

The HMP Watchdog Agent provides two services: ILOM watchdog and host watchdog.

You can configure the HMP Watchdog Agent using the `ilomconfig` CLI command (preferred), or by editing the `hmp_watchdogd.conf` file. For instructions, see "Configuring the HMP Watchdog Agent" on page 48.

For information on how to install this component, refer to the *Oracle Hardware Management Pack Installation Guide*.

Configuring Hardware Management Agent and Hardware SNMP Plugins

This section provides instructions for configuring the Hardware Management Agent and Hardware SNMP Plugins, as well as information about using Hardware Management Agent successfully. The section contains the following:

Hardware Management Agent Configuration File

The Hardware Management Agent records log messages in the `hwmgmtd.log` file. These messages can be used to troubleshoot the running status of the Hardware Management Agent.

To configure the level of detail stored, set the `hwagentd_log_levels` parameter in the `hwmgmtd.conf` file. See "Configure the Hardware Management Agent Logging Level" on page 19.

You can use two versions of this parameter; *basic* or *bit flags*. The following subsections describe each.

Basic Parameters

The following table shows the basic values for the `hwagentd_log_levels` parameter in the `hwmgmtd.conf` file.

Log Level	Messages Logged
ERROR	Any error messages generated by the Hardware Management Agent
WARNING	Any error and warning messages generated by the Hardware Management Agent
INFO	Any error and warning messages generated by the Hardware Management Agent and informative messages about normal functioning

Bit Flags Parameters

Using bit flags allows you to set the logging level with a finer level of granularity. The following table shows the values.

Note - It is recommended to use the logging levels above. The bit flags options are for advanced troubleshooting.

Log Level	Bit Code	Messages Logged
EMERG	0x0001	Information about the system being unusable
ALARM	0x0002	Information about any immediate action that must be taken
CRIT	0x0004	Information related to the Hardware Management Agent either not starting or stopping because of critical conditions
ERROR	0x0008	Information about any error messages generated by the Hardware Management Agent
WARNING	0x0010	Information about any error and warning messages generated by the Hardware Management Agent
NOTICE	0x0020	Information related to normal functioning
INFO	0x0040	Information about any error and warning messages generated by the Hardware Management Agent and informative messages about normal functioning
DEBUG	0x0080	Verbose debug-level messages, useful in troubleshooting
TRACE	0x0100	Highly verbose debug-level messages, useful in troubleshooting

Note - levels DEBUG and TRACE generate a lot of detailed messages and are designed for troubleshooting. These levels are not recommended for production usage.

For example, when you want to set all logging levels between EMERG and NOTICE, the bit code values of all the required levels must be added and then converted to a decimal value. Referring to preceding table, the addition would be as follows:

0x0001 + 0x0002 + 0x0004 + 0x0008 + 0x0010 + 0x0020 = 0x003f

Converting 0x003f to decimal equals 63, which is the desired log level. This is the decimal number that should be assigned to the `hwagentd_log_levels` parameter in the `hwmgmtd.conf` file.

▼ Configure the Hardware Management Agent Logging Level

1. **Find the `hwmgmtd.conf` file and open it for editing.**

   The following table shows the file location on different operating systems.

Operating System	Configuration file path
Oracle Solaris	`/etc/opt/`*ssm_directory*`/hwmgmtd.conf` Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.
Linux based	`/etc/`*ssm_directory*`/hwmgmtd.conf` Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.
Microsoft Windows	<*Program Files*>`\Oracle\Oracle Hardware Management Pack\conf\hwmgmtd.conf`

2. **Find the `hwagentd_log_levels` parameter and change the logging level to one of the options described in the previous subsections.**

3. **Save the modified `hwmgmtd.conf` file.**

4. **Choose one of the following options to make the Hardware Management Agent reread the `hwmgmtd.conf` file:**

   - **On Oracle Solaris refresh Hardware Management Agent.**

     This forces the OS to reread the `hwmgmtd.conf` file.

     ```
     /usr/sbin/svcadm disable svc:/application/management/hwmgmtd:default
     ```

     ```
     /usr/sbin/svcadm enable svc:/application/management/hwmgmtd:default
     ```

   - **On Linux restart Hardware Management Agent.**

     This forces Linux to reread the `hwmgmtd.conf` file.

     ```
     /sbin/service hwmgmtd restart
     ```

- **On Windows restart the service using the Microsoft Management Console Services snap-in.**

The Hardware Management Agent rereads the `hwmgmtd.conf` file with the modified `hwagentd_log_levels` parameter.

Configuring Your Host Operating System's SNMP

The Hardware Management Agent uses native SNMP for network communications. This means that SNMP must already be installed and running on the host operating system. For the Hardware Management Agent to be able to use SNMP, you must ensure that SNMP is configured correctly. Incorrect settings can cause the Hardware Management Agent to have limited, or no, network connectivity.

Configuration details are operating system specific, refer to the following for more information:

- For Oracle Solaris and Linux based operating systems, the `snmpd.conf` file controls network access to the Hardware Management Agent. See "(Solaris and Linux) Configuring Net-SNMP/SMA" on page 20.
- For Windows operating systems, the SNMP service controls network access to the Hardware Management Agent. See "(Windows) Configuring SNMP" on page 23.

(Solaris and Linux) Configuring Net-SNMP/SMA

Depending on which operating system the Hardware Management Agent has been installed on, you can find the `snmpd.conf` file at the path shown in the following table.

Operating System	Path to `snmpd.conf`
Linux	`/etc/snmp/snmpd.conf`
Oracle Solaris 10 Operating System	`/etc/sma/snmp/snmpd.conf`
Oracle Solaris 11 Operating System	`/etc/net-snmp/snmp/snmpd.conf`

The exact modifications you need to make to the `snmpd.conf` file depend on which host operating system the Hardware Management Agent is running on. The following procedures explain how to configure SNMP gets, sets, and traps.

Note - the following instructions assume you are using an unmodified `snmpd.conf` file. If you have customized your `snmpd.conf` file, use these instructions as a guide to make sure your `snmpd.conf` file is compatible with the Hardware Management Agent.

This section covers the following procedures:

How to Configure SNMP Gets

SNMP gets enable you to read data filled by the Hardware Management Agent.

To be able to perform SNMP gets, use the following procedure.

1. **Open your `snmpd.conf` file for editing.**
2. **Choose one of the following options:**

   - **For Red Hat Enterprise Linux, add the following line to `snmpd.conf`:**

     ```
     view systemview included .1.3.6.1.4.
     ```

     This adds the Hardware SNMP Plugins to the specified view.

   - **For Oracle Solaris OS and SUSE Linux Enterprise Server, add the following line to `snmpd.conf`:**

     ```
     rocommunity public
     ```

     This adds a read-only community from a network location other than localhost.

How to Configure SNMP Sets

Use the following procedure to enable SNMP to perform sets.

1. **Open your `snmpd.conf` file for editing.**
2. **Choose one of the following options:**

   - **For Oracle Solaris and SUSE Linux Enterprise Server add the following line:**

     ```
     rwcommunity private
     ```

     By default the public community is blocked as rocommunity on these operating systems.

   - **For Red Hat Enterprise Linux, change:**

     ```
     access notConfigGroup "" any noauth exact systemview none none
     ```

     to the following:

     ```
     access notConfigGroup "" any noauth exact systemview systemview none
     ```

     This modification grants write access for the specified view and group. In this example the specified view is `systemview` and the specified group is `NotConfigGroup`. By default, the group uses the public community string.

How to Configure SNMP Traps

1. **Open your `snmpd.conf` file for editing.**
2. **Depending on the version of SNMP traps you want to send:**

   - **To be able to send SNMP version 1 traps from the Hardware Management Agent, add the following line to `snmpd.conf`:**

     `trapsink` *host communitystring trapport*

   - **To be able to send SNMP version 2 traps from the Hardware Management Agent, add the following line to `snmpd.conf`:**

     `trap2sink` *host communitystring trapport*

Example 1 Setting SNMP Version 2 Traps

The following example shows the line added to the `snmpd.conf` file to configure SNMP Traps using SNMP version 2:

```
trap2sink 10.18.141.22 public 162
```

(Windows) Configuring SNMP

On Windows operating systems there is not a `snmpd.conf` file. You configure the SNMP service in the Windows Microsoft Management Console Services snap-in.

▼ (Windows) How to Configure SNMP

1. **From the Start menu Administrative Tools option, select Services.**

   The Microsoft Management Console Services snap-in opens.

2. **Double-click the SNMP service.**

   The SNMP service options open.

3. **In the SNMP service options, select the Security tab.**

   Configure the community rights.

4. **In the SNMP service options, select the Traps tab.**

   Configure the destination you want to send SNMP traps to.

5. **Close the SNMP service options.**

Oracle Server Hardware SNMP Plugins Overview

This section contains overviews of the Management Information Bases (MIBs) that are implemented by Oracle Server Hardware SNMP Plugins. This section contains the following:

Overview of Sun HW Monitoring MIB

The Sun HW Monitoring Management Information Base (MIB) provides the following details about the server or server module implementing this MIB:

- A hardware inventory of all Field Replaceable Units (FRU) and sensors monitoring different physical parameters
- Parent/child relationship or containment information of all FRUs and sensors
- Individual status of each sensor as well as combined status of each device type
- Any threshold values configured for each sensor, where applicable
- Details about the service processor
- Information about total power consumption

The MIB is subdivided into sections, based on the information provided by the MIB objects. The information provided by the MIB objects is categorized into logically divided groups of scalars, as well as MIB tables.

For a complete list of all of the objects defined by each group, refer to the comments section defined at the beginning of each group in the `SUN-HW-MONITORING-MIB.mib` file.

The following sections briefly describe each of the MIB sections, with some examples of the objects defined in each group:

Sun Server Product and Chassis

The first two groups, sunHwMonProductGroup and sunHwMonProductChassisGroup, define scalar MIB objects that provide information about the server, including part number, and manufacturer. These groups are:

- sunHwMonProductGroup is a scalar group that provides general product details about the server or server module, such as the part number, type, name, and serial number.
- sunHwMonProductChassisGroup is a scalar group that provides details about the server's chassis or the chassis into which the server has been inserted.

Note - sunHwMonProductChassisGroup is populated only on server modules, where it is relevant.

Sun Server Service Processor

The Sun Server Service Processor group consists of one group, sunHwMonSPGroup, which is a scalar group that provides details about the server's Oracle Integrated Lights Out Management (ILOM) service processor. This group includes information such as serial number, manufacturer, MAC Address, IP details, and Web accessibility information such as the URL to access the Oracle ILOM Web interface.

Sun Server Hardware Monitoring MIB

The Sun Server Hardware Monitoring MIB group consists of one scalar group, sunHwMonMibGroup that provides details about the SUN-HW-MONITORING-MIB itself, such as MIB version number.

Sun Server Hardware Management Agent

The Sun Servers Hardware Management Agent group consists of one scalar group, sunHwMonAgentSoftwareGroup that provides details about the Hardware Management Agents associated with this MIB, such as the version of the Agent and the connection status to Oracle ILOM.

Sun Server Hardware Inventory

The Sun Servers Hardware Inventory group consists of one scalar group, sunHwMonInventoryGroup with a MIB table, sunHwMonInventoryTable. This table contains details about the server's field replaceable units (FRUs). For each FRU, it includes the name, type, description, part number, status, and the FRU in which it is contained (if any).

Sun Server Hardware Monitor Sensor Group

The sunHwMonSensorGroup contains details about all of the server's hardware sensors, except indicators. The MIB objects that define the sensor properties are hierarchically and logically grouped based on device type, for example temperature or voltage, as well as sensor type, for example numeric or discrete.

The sunHwMonSensorGroup also contains a device-specific group for all significant device types, such as sunHwMonVoltageGroup or sunHwMonCurrentGroup. There is also a group for sensors that are not part of any device—specific group.

Each of the groups listed below contains two tables. One table provides details about all of the numeric sensors of this device type and the other table provides details about all of the discrete sensors of corresponding device type on the server.

The numeric sensors tables provide details about numeric sensors such as the sensor name, sensor type, the current reading, defined thresholds, current status, perceived severity, and the FRU in which the sensor is contained. The discrete sensors tables provide details about discrete sensors, such as sensor name, sensor type, sensor state, perceived severity, and the FRU in which the sensor is contained.

The alarm status of an entity can be one of the following, where critical is the most severe and indeterminate is the least severe.

- critical
- major

- minor
- warning
- cleared
- indeterminate

The sunHwMonSensorGroup contains the following groups:

- sunHwMonSensorAlarmStatusGroup is a scalar group that provides a single view of the alarm status of the server and aggregate status per device type such as rolled-up status of all voltage sensors. This is the main value used to obtain the overall status of a server. The individual sensor status is provided by MIB objects that are defined in the corresponding device-specific group.
- sunHwMonVoltageGroup contains two MIB tables that provide details regarding all voltage sensors contained in the server.
- sunHwMonCurrentGroup contains two MIB tables that provide details regarding all current sensors contained in the server.
- sunHwMonPowerDeviceGroup contains two MIB tables that provide details regarding all power device sensors contained in the server.
- sunHwMonCoolingDeviceGroup contains two MIB tables that provide details regarding all cooling device sensors contained in the server.
- sunHwMonTemperatureGroup contains two MIB tables that provide details regarding all temperature sensors contained in the server.
- sunHwMonMemoryGroup contains two MIB tables that provide details regarding all memory sensors contained in the server.
- SunHwMonProcessorGroup contains two MIB tables that provide details regarding all processor sensors contained in the server.
- sunHwMonHardDriveGroup contains two MIB tables that provide details regarding all hard drive sensors contained in the server.
- sunHwMonIOGroup contains two MIB tables that provide details regarding all input/output sensors contained in the server.
- sunHwMonSlotOrConnectorGroup contains two MIB tables that provide details regarding all slot or connector sensors contained in the server.
- sunHwMonOtherSensorGroup contains two MIB tables that provide details regarding all sensors contained in the server that are not part of above defined device type groups.

sunHwMonIndicatorGroup

This group contains multiple groups that provide details about the indicators present on the server. These groups are as follows:

- sunHwMonIndicatorLocator is a scalar group that provides details about the locator indicator, such as the name of the locator indicator sensor and its status. The sunHwMonIndicatorLocatorCurrentStatus MIB object is a read-write MIB object. You can control the locator indicator sensor through an SNMP set command, using a community string with write access.
- sunHwMonIndicatorService is a scalar group that provides the name and status of the service indicator sensor.
- sunHwMonIndicatorAll contains sunHwMonIndicatorTable, which provides details about all indicators present on the server, such as power supply failure indicator or fan failure indicator.

sunHwMonTotalPowerConsumption

This scalar group provides details about the server's total power consumption, including:

- Sensor name and type
- Current reading
- Defined thresholds
- Current status
- Perceived severity
- The FRU in which the sensor is contained

Note - Data is available here only if the platform has implemented a total power consumption indicator.

Overview of Sun HW Trap MIB

The Hardware Management Agent uses the Sun HW Trap MIB to implement SNMP traps. These traps report the environmental state of the server as well as faults, errors, and other conditions affecting hardware components.

The SNMP traps are categorized into three groups.

- Any SNMP trap name ending in Ok or Error, as well as any SNMP trap name containing Threshold, is reporting a change in a sensor value.
- Any SNMP trap name ending in Fault is reporting a problem detected by the system's fault management subsystem, if such a subsystem is available on the server.

- The final group is the status SNMP traps, which report the environmental state and any hardware information that is not covered by the two previous groups.

For more detailed information on the Sun HW Trap MIB, see the comments in the `SUN-HW-TRAP-MIB.mib` file.

Overview of Sun Storage MIB

The Sun Storage MIB supplements the Sun HW Monitoring MIB with storage-related information. The following sections briefly describe each of the MIB sections:

- "Sun Storage MIB Objects" on page 30
- "Physical and Logical Storage Objects" on page 30

Sun Storage MIB Objects

The following scalar objects contain information about the Sun Storage MIB itself:

- sunStorageAgentVersion defines the version of the software implementing the sunStorageMIB. The version is in the following format: *MajorVersion.MinorVersion.SubMinorVersion* (for example: 1.2.3).
- sunStorageMibVersion defines the version of the SUN-STORAGE-MIB this agent implements. The version defined is in the format of *MajorVersion.MinorVersion.SubMinorVersion* (for example: 1.3.0).

Physical and Logical Storage Objects

The following tables list physical and logical storage objects:

- sunStorageControllerTable. The storage controller object represents either an on-board or bus-attached storage controller. The properties associated with a controller object describe the type of controller (vendor and model) as well as the features it supports (such as RAID). The table is indexed with an arbitrary integer to uniquely identify each entry. Entries can contain the following:
  - Identifying: name, part number, serial number, manufacturer, model, firmware version, and PCIbus address
  - RAID capabilities: levels supported, maximum volumes manageable, number of spares, and stripe size

    - Status: operational and alarm
- sunStorageDiskTable. Each disk object corresponds to one physical disk that is available to the host operating system. Entries in this table might have parent objects in other tables (such as sunStorageControllerTable). The table is indexed with sunHwMonFruIndex, so that information corresponding to the same physical disk is retrievable from both the sunHwMonInventoryTable and sunStorageDiskTable at the same index.
    - Identifying: name and OS device name
    - Relational: parent name and index, slot number
    - Descriptive: physical type, interface type, and capacity
    - Status: mapping, RAID, and operational
- Entries can contain the following:
- sunStorageVolumeTable. This table contains logical volume objects that correspond to a logical disk visible to the host OS. Only RAID logical volumes are supported. The table is indexed with an arbitrary integer to uniquely identify each entry. Entries can contain the following:
    - Identifying: name, OS device name, and mount point
    - Relational: parent name and index
    - Descriptive: capacity, RAID level, and sizing
    - Status: mapping, mounting, RAID parameters, task, and operational
- sunStorageLogicalCompTable. A logical component node represents an active or passive component of its logical device parent. A logical component object is always a direct child of a logical device node. In the case of a RAID logical device, the logical component represents a physical device, or part of a physical device, used to create the specified RAID level. The table is indexed with an arbitrary integer to uniquely identify each entry. Entries can contain the following:
    - Identifying: name, disk name, and index
    - Relational: parent name and index
    - Status: RAID spare and RAID operational

Working With Management Agents

Once the Management Agents are installed on your Oracle Server, you can monitor the server. The Hardware Management Agent provides the SNMP Plugins layer, which enables you to retrieve and set information using SNMP, and to generate SNMP traps.

This section provides the following:

Retrieving and Setting Information Through SNMP

The following section provides some examples of using Net-SNMP's snmpwalk utility to get and set information from Oracle servers running the Hardware Management Agent. For more information on the Hardware Management Agent functionality shown here, see "Overview of Sun HW Monitoring MIB" on page 25 or the `SUN-HW-MONITORING-MIB.mib` file.

The format of the Net-SNMP snmpwalk command is:

`snmpwalk` *Application options Common Options OID*

The command examples in this section use the default MIB locations, which are listed below. If your MIB files are located somewhere else, change the pathnames accordingly.

- Solaris 11.1: `/usr/lib/`*ssm_directory*`/lib/mibs`

Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

- Solaris 10: `/opt/`*ssm_directory*`/lib/mibs`

  Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.
- Linux: `/usr/share/snmp/mibs`
- Windows: Default HMP installation directory `/mib`

For more information, see the Net-SNMP documentation.

sunHwMonProductGroup

The sunHwMonProductGroup contains information about the server implementing the MIB.

The following procedures are covered in this section:

- "Retrieve the Product Information from a Sun x86 Server" on page 34
- "Retrieve The Product Information on a Sun x86 Server Module" on page 35

▼ Retrieve the Product Information from a Sun x86 Server

- **At the command prompt, type the following:**

  - On a Linux system:

    ```
    snmpwalk -v 2c -c public -M /usr/share/snmp/mibs/ -m ALL localhost
    sunHwMonProductGroup
    ```
  - On an Oracle Solaris 11.1 system:

    ```
    snmpwalk -v 2c -c public -M +/usr/lib/ssm_directory/lib/mibs/ -m ALL localhost
    sunHwMonProductGroup
    ```

    Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.
  - On an Oracle Solaris 10 system:

    ```
    snmpwalk -v 2c -c public -M +/opt/ssm_directory/lib/mibs/ -m ALL localhost
    sunHwMonProductGroup
    ```

Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonProductName.0 = STRING: SUN FIRE X4440

SUN-HW-MONITORING-MIB::sunHwMonProductType.0 = INTEGER: rackmount(3)

SUN-HW-MONITORING-MIB::sunHwMonProductPartNumber.0 = STRING: 602-4058-01

SUN-HW-MONITORING-MIB::sunHwMonProductSerialNumber.0 = STRING: 0823QBU01C

SUN-HW-MONITORING-MIB::sunHwMonProductManufacturer.0 = STRING: SUN MICROSYSTEMS

SUN-HW-MONITORING-MIB::sunHwMonProductSlotNumber.0 = INTEGER: -1

SUN-HW-MONITORING-MIB::sunHwMonProductUUID.0 = STRING:
080020FFFFFFFFFFFFFF00144FEDE5E0

SUN-HW-MONITORING-MIB::sunHwMonProductBiosVersion.0 = STRING: S90_3B18
```

Note - On a Sun x86 rack mount server, the following line signifies that there is no slot number (nodef).

```
sunHwMonProductSlotNumber.0 = INTEGER: -1
```

This is expected behavior because slot numbers are relevant only to blade servers. Rackmount servers do not have slot numbers.

▼ Retrieve The Product Information on a Sun x86 Server Module

- **At the command prompt, type the following:**

  - On a Linux system:

    ```
    snmpwalk -v 2c -c public -M /usr/share/snmp/mibs/ -m ALL localhost
    sunHwMonProductGroup
    ```

  - On an Oracle Solaris 11.1 system:

    ```
    snmpwalk -v 2c -c public -M +/usr/lib/ssm_directory/lib/mibs/ -m ALL localhost
    sunHwMonProductGroup
    ```

    Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

- On an Oracle Solaris 10 system:

  ```
  snmpwalk -v 2c -c public -M +/opt/ssm_directory/lib/mibs/ -m ALL localhost
  sunHwMonProductGroup
  ```

  Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonProductName.0 = STRING: Sun Blade X6250 Server
Module

SUN-HW-MONITORING-MIB::sunHwMonProductType.0 = INTEGER: blade(4)

SUN-HW-MONITORING-MIB::sunHwMonProductPartNumber.0 = STRING: 540-7254-01

SUN-HW-MONITORING-MIB::sunHwMonProductSerialNumber.0 = STRING: 142300943223

SUN-HW-MONITORING-MIB::sunHwMonProductManufacturer.0 = STRING: Sun Microsystems
Inc

SUN-HW-MONITORING-MIB::sunHwMonProductSlotNumber.0 = INTEGER: 1

SUN-HW-MONITORING-MIB::sunHwMonProductUUID.0 = STRING:
080020FFFFFFFFFFFFFF001B24782F9C

SUN-HW-MONITORING-MIB::sunHwMonProductBiosVersion.0 = STRING: S90_3B18
```

sunHwMonProductChassisGroup

This group is filled only on Sun x86 server modules and represents the chassis holding the server module.

▼ Retrieve the Server Module's Product Chassis Information

- **At the command prompt, type the following:**

  - On a Linux system:

    ```
    snmpwalk -v 2c -c public -M /usr/share/snmp/mibs/ -m ALL localhost
    sunHwMonProductChassisGroup
    ```

  - On an Oracle Solaris 11.1 system:

```
snmpwalk -v 2c -c public -M +/usr/lib/ssm_directory/lib/mibs/ -m ALL localhost
sunHwMonProductChassisGroup
```

Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

- On an Oracle Solaris 10 system:

```
snmpwalk -v 2c -c public -M +/opt/ssm_directory/lib/mibs/ -m ALL localhost
sunHwMonProductChassisGroup
```

Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonProductChassisName.0 = STRING: SUN BLADE 6000
MODULAR SYSTEM

SUN-HW-MONITORING-MIB::sunHwMonProductChassisPartNumber.0 = STRING: 541-1983-07

SUN-HW-MONITORING-MIB::sunHwMonProductChassisSerialNumber.0 = STRING: 1005LCB-
0728YM01R7

SUN-HW-MONITORING-MIB::sunHwMonProductChassisManufacturer.0 = STRING: SUN
MICROSYSTEMS
```

sunHwMonSPGroup

This group contains information about the Oracle ILOM service processor.

▼ Retrieve Service Processor Information

- **At the command prompt, type the following:**

  - On a Linux system:

```
snmpwalk -v 2c -c public -M /usr/share/snmp/mibs/ -m ALL localhost
sunHwMonSPGroup
```

  - On an Oracle Solaris 11.1 system:

```
snmpwalk -v 2c -c public -M +/usr/lib/ssm_directory/lib/mibs/ -m ALL localhost
sunHwMonSPGroup
```

Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

- On an Oracle Solaris 10 system:

  ```
  snmpwalk -v 2c -c public -M +/opt/ssm_directory/lib/mibs/ -m ALL localhost
  sunHwMonSPGroup
  ```

  Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonSPSerialNumber.0 = STRING: 1762TH1-0750000707
SUN-HW-MONITORING-MIB::sunHwMonSPManufacturer.0 = STRING: ASPEED
SUN-HW-MONITORING-MIB::sunHwMonSPFWVersion.0 = STRING: 2.0.3.10
SUN-HW-MONITORING-MIB::sunHwMonSPMacAddress.0 = STRING: 0:1b:24:78:2f:a1
SUN-HW-MONITORING-MIB::sunHwMonSPIPAddress.0 = IpAddress: 10.18.141.164
SUN-HW-MONITORING-MIB::sunHwMonSPNetMask.0 = IpAddress: 255.255.255.128
SUN-HW-MONITORING-MIB::sunHwMonSPDefaultGateway.0 = IpAddress: 10.18.141.129
SUN-HW-MONITORING-MIB::sunHwMonSPIPMode.0 = INTEGER: dhcp(2)
SUN-HW-MONITORING-MIB::sunHwMonSPURLToLaunch.0 = STRING:
SUN-HW-MONITORING-MIB::sunHwMonSPSystemIdentifier.0 = STRING:
```

Note - When using Oracle ILOM 2.0 the following lines are returned:

```
SUN-HW-MONITORING-MIB::sunHwMonSPURLToLaunch.0 = STRING:

SUN-HW-MONITORING-MIB::sunHwMonSPSystemIdentifier.0 = STRING:
```

This is expected behavior because this information is specific to Oracle ILOM 3.0.

sunHwMonInventoryTable

Information about only one FRU, `mb.net0.fru`, is shown in this example.

▼ Retrieve Inventory Information

- **At the command prompt, type the following:**

  - On a Linux system:

    ```
    snmpwalk -v 2c -c public -M /usr/share/snmp/mibs/ -m ALL localhost
    sunHwMonInventoryTable | grep '.148 = '
    ```

  - On an Oracle Solaris 11.1 system:

    ```
    snmpwalk -v 2c -c public -M +/usr/lib/ssm_directory/lib/mibs/ -m ALL localhost
    sunHwMonInventoryTable | grep '.148 = '
    ```

    Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

  - On an Oracle Solaris 10 system:

    ```
    snmpwalk -v 2c -c public -M +/opt/ssm_directory/lib/mibs/ -m ALL localhost
    sunHwMonInventoryTable | grep '.148 = '
    ```

    Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

  where `grep '.148 = '` is filtering for results with a property of the FRU we are interested in.

  You should see output similar to the following:

  ```
  SUN-HW-MONITORING-MIB::sunHwMonFruName.148 = STRING: /SYS/MB/NET0
  SUN-HW-MONITORING-MIB::sunHwMonFruType.148 = INTEGER: networkInterface(80)
  SUN-HW-MONITORING-MIB::sunHwMonFruDescr.148 = STRING:
  SUN-HW-MONITORING-MIB::sunHwMonFruPartNumber.148 = STRING: 82546GB
  SUN-HW-MONITORING-MIB::sunHwMonFruSerialNumber.148 = STRING: 00:14:4F:A8:39:44
  SUN-HW-MONITORING-MIB::sunHwMonFruManufacturer.148 = STRING:
  SUN-HW-MONITORING-MIB::sunHwMonFruStatus.148 = INTEGER: indeterminate(6)
  SUN-HW-MONITORING-MIB::sunHwMonParentFruIndex.148 = INTEGER: 146
  SUN-HW-MONITORING-MIB::sunHwMonParentFruName.148 = STRING: /SYS/MB
  ```

Note - When using Oracle ILOM 2.0 the following lines are returned:

```
SUN-HW-MONITORING-MIB::sunHwMonFruType.75 = INTEGER: unknown(1)

SUN-HW-MONITORING-MIB::sunHwMonParentFruIndex.75 = INTEGER: -1

SUN-HW-MONITORING-MIB::sunHwMonParentFruName.75 = STRING:
```

This is expected behavior because this information is specific to Oracle ILOM 3.0. In this case, the -1 signifies nodef.

sunHwMonSensorGroup

In the following example, the numeric sensor MB/V_+12V is retrieved.

▼ Retrieve the Sensor Group Information

- **At the command prompt, type the following:**

  - On a Linux system:

    ```
    snmpwalk -v 2c -c public -M /usr/share/snmp/mibs/ -m ALL localhost
    sunHwMonSensorGroup | grep '\.9 = '
    ```

  - On an Oracle Solaris 11.1 system:

    ```
    snmpwalk -v 2c -c public -M +/usr/lib/ssm_directory/lib/mibs/ -m ALL localhost
    sunHwMonSensorGroup | grep '\.9 = '
    ```

    Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

  - On an Oracle Solaris 10 system:

    ```
    snmpwalk -v 2c -c public -M +/opt/ssm_directory/lib/mibs/ -m ALL localhost
    sunHwMonSensorGroup | grep '\.9 = '
    ```

    Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

where `grep '\.9 = '` is filtering a property of the FRU we are interested in.

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorType.9 = INTEGER: voltage
(133)

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorName.9 = STRING: /SYS/MB/V_
+12V

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorParentFruIndex.9 = INTEGER:
146

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorParentFruName.9 =
STRING: /SYS/MB

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorAlarmStatus.9 = INTEGER:
cleared(1)

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorStateDescr.9 = STRING: Normal

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorCurrentValue.9 = INTEGER:
12160

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorBaseUnit.9 = INTEGER: volts
(4)

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorExponent.9 = INTEGER: -3

SUN-HW-MONITORING-MIB::
sunHwMonNumericVoltageSensorUpperNonRecoverableThreshold.9 = INTEGER: 14994

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorUpperCriticalThreshold.9 =
INTEGER: 13986

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorUpperNonCriticalThreshold.9 =
INTEGER: 12978

SUN-HW-MONITORING-MIB::
sunHwMonNumericVoltageSensorLowerNonRecoverableThreshold.9 = INTEGER: 8946

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorLowerCriticalThreshold.9 =
INTEGER: 9954

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorLowerNonCriticalThreshold.9 =
INTEGER: 10962

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorEnabledThresholds.9
= BITS: FC lowerThresholdNonCritical(0) upperThresholdNonCritical(1)
lowerThresholdCritical(2) upperThresholdCritical(3) lowerThresholdFatal(4)
upperThresholdFatal(5)
```

Note - When using Oracle ILOM 2.0 the following lines are returned:

```
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorType.9 = INTEGER: unknown(1)

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorParentFruIndex.9 = INTEGER:
-1

SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorParentFruName.9 = STRING:
```

This is expected behavior because this information is specific to Oracle ILOM 3.0.

Tip - When analyzing the following lines, do not forget that the sunHwMonNumericVoltageSensorCurrentValue is returned using the exponent set in sunHwMonNumericVoltageSensorExponent.

```
SUN-HW-MONITORING-MIB::

sunHwMonNumericVoltageSensorCurrentValue.9 = INTEGER: 12290

SUN-HW-MONITORING-MIB::

sunHwMonNumericVoltageSensorBaseUnit.9 = INTEGER: volts(4)

SUN-HW-MONITORING-MIB::

sunHwMonNumericVoltageSensorExponent.9 = INTEGER: -3
```

This example has an exponent of -3, which means that the voltage value of sunHwMonNumericVoltageSensorCurrentValue has to be multiplied by 10^{-3}, resulting in 12.290 volts.

sunHwMonIndicatorLocator

You can get and set the sunHwMonIndicatorLocator. The following example sets the sunHwMonIndicatorLocator to integer(i) value 7, which means fastBlink for this OID.

▼ Set the Indicator Locator

- **At the command prompt, type the following:**

```
# snmpset -v2c -c public -m ALL localhost SUN-HW-MONITORING-MIB::
sunHwMonIndicatorLocatorCurrentStatus.0 i 7
```

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonIndicatorLocatorCurrentStatus.0 = INTEGER:
fastBlinking(7)
```

Generating SNMP Traps

The combination of Hardware Management Agent and Hardware SNMP Plugins enables you to generate SNMP traps. To test this, you can use IPMItool, which is a component of Hardware Management Pack, to inject a simulated fault. This causes the Hardware SNMP Plugins to generate an SNMP fault.

▼ Inject a Simulated Fault

Caution - This procedure returns test SNMP traps, however the values received might not match the values you expect to see when a real SNMP trap is generated. This does not impact non-test SNMP trap functionality.

1. **At the command prompt, type:**

   `ipmitool -U` *user* `-P` *password* `-H` *hostname* `-v sdr list`

   Choose a sensor from the returned list that you want to inject a simulated fault to. In this example the IPMI event: 'P0/VTT' unc assert is used.

2. **At the command prompt, type:**

   `# ipmitool -U` *user* `-P` *password* `-H` *hostname* `event 'P0/VTT' unc assert`

   This injects the IPMI event: 'P0/VTT' unc assert.

   You should receive an SNMP trap similar to the following:

   ```
   sysUpTime.0 = Timeticks: (4300) 0:00:43.00

   snmpModules.1.1.4.1.1 = OID: sunHwTrapVoltageNonCritThresholdExceeded

   sunHwTrapSystemIdentifier.0 = STRING: sg-prg-x6220-01-sp0

   sunHwTrapChassisId.0 = STRING: 1005LCB-0728YM01R7::0739AL71EA
   ```

```
sunHwTrapProductName.0 = STRING: SUN BLADE 6000 MODULAR SYSTEM::SUN BLADE X6220
SERVER MODULE

sunHwTrapComponentName.0 = STRING: /SYS/MB/P0/VTT

sunHwTrapThresholdType.0 = INTEGER: upper(1)

sunHwTrapThresholdValue.0 = STRING:

sunHwTrapSensorValue.0 = STRING:

sunHwTrapAdditionalInfo.0 = STRING: Upper Non-critical going high

sunHwTrapAssocObjectId.0 = OID: zeroDotZero

sunHwTrapSeverity.0 = INTEGER: nonCritical(4)
```

You can verify the SNMP trap by checking the syslog record, which should contain something similar to the following:

```
sg-prg-x6250-01 hwagentd[3470]: P0/VTT (Sensor ID: 0x1b) (Record ID: 0x821):
Upper Non-critical going high.
```

The messages stored in syslog or the Windows application log correspond exactly to the SNMP traps. On Linux and Oracle Solaris operating systems, the messages are logged with facility `daemon` and level `notice`.

Note - If records corresponding to SNMP traps are not being stored on Linux and Oracle Solaris operating systems, make sure that the `daemon` facility and `notice` level are enabled.

Configuring the HMP Watchdog Agent

Starting with Oracle Hardware Management Pack (HMP) 2.3.0.0, the HMP watchdog agent and associated ILOM and host watchdog services are available for systems running a Linux OS. The following sections describe the agent and how to configure it. For information about using the agent, see:

HMP Watchdog Agent Overview

When it is installed and configured, the HMP Watchdog Agent periodically checks the host and/or Oracle ILOM and performs a user-configured action if either proves unresponsive. The actions can include posting a warning to a log file, resetting the corresponding device, and in the case of the host, power cycling or powering off the host.

The Oracle HMP watchdog agent is an optional Oracle HMP component that you can install using the Oracle HMP installer. For information on how to install this component, refer to the *Oracle Hardware Management Pack Installation Guide*.

Note the following important information regarding the HMP watchdog agent:

- Your system must meet the following requirements to run the HMP watchdog agent:
  - A Linux operating system installed
  - Oracle HMP 2.3.0.0 or later installed
  - Oracle ILOM 3.2.2 or later on the SP
- The agent must be started after it is installed and it must be restarted after the host is reset..
- The host or ILOM watchdog configuration is preserved after the host is reset.
- You can configure the HMP watchdog agent using the `ilomconfig` CLI command, or by editing the `hmp_watchdogd.conf` file. For instructions, see "Configuring the HMP Watchdog Agent" on page 48. The `ilomconfig` is the preferred method.

The HMP watchdog agent provides two services, ILOM watchdog and host watchdog.

ILOM Watchdog Overview

The ILOM watchdog periodically queries Oracle ILOM. If Oracle ILOM becomes unresponsive, the host either posts a warning or resets Oracle ILOM.

It also logs an appropriate message into the HMP watchdog log file, and in the host's system log at `/var/log/messages`.

You can control ILOM watchdog's parameters from the `ilomconfig` command, or by editing the HMP watchdog agent's configuration file. See "ILOM Watchdog Parameters" on page 47 for more information on the ILOM watchdog parameters that you can modify. The `ilomconfig` command is the preferred method for controlling ILOM watchdog.

Host Watchdog Overview

The host watchdog causes Oracle ILOM to watch the host. If the host becomes unresponsive, Oracle ILOM performs a customer-configured action, which can be: Warning, Reset, Power Off, or Power Cycle.

You can control the host watchdog parameters using the `ilomconfig` command, or by editing the HMP watchdog agent's configuration file. The `ilomconfig` command is the preferred method for controlling host watchdog.

See "Host Watchdog Parameters" on page 48 for more information on the host watchdog parameters that you can modify.

Host watchdog is built on top of standard IPMI watchdog timer capability. The host watchdog interacts with the IPMI watchdog timer as follows:

- When the host watchdog is enabled by the user, it first checks to see if the IPMI watchdog timer is already started. If the IPMI watchdog timer is started, the host watchdog issues a log message indicating that the watchdog timer is already started, and the host watchdog remains in the Disable state.
- When the host watchdog has already been enabled, the host watchdog periodically resets the IPMI watchdog timer to the configuration values of host watchdog. This takes care of the case where someone changes configuration outside of HMP watchdog agent.
- When the OS hangs, the IPMI watchdog timer is not reset by the time the timer expires. This causes Oracle ILOM to perform the action specified by timer-action parameter.

HMP Watchdog Agent Parameters

The following sections describe the parameters that you can set for the HMP watchdog agent:

- "ILOM Watchdog Parameters" on page 47
- "Host Watchdog Parameters" on page 48

ILOM Watchdog Parameters

The ILOM watchdog parameters can be set by the user in one of the following ways:

- Preferred method: Use the `ilomconfig` commands to change the parameters. See "Configure ILOM Watchdog Using `ilomconfig` Commands" on page 51. For more information on ilomconfig CLI commands, see "Host Watchdog ilomconfig Commands" in *Oracle Server CLI Tools User's Guide*.
- Manually edit the configuration file. See "Manually Edit the Configuration File" on page 52.

The following table lists the ILOM watchdog parameters.

Parameter Name	Parameter in Configuration File	Description	Possible Values	Default value
Query Time Interval	`ilom_watchdog_query_time_interval`	The interval at which the host queries Oracle ILOM (seconds). If Oracle ILOM does not respond the host performs the Timer Action.	Positive integer up to 214748364	60 seconds
Timer Action	`ilom_watchdog_timer_action`	The action which the host performs when it notices that Oracle ILOM is unresponsive	`Warning|Reset`	Warning
Number of Consecutive SP Resets	`ilom_watchdog_number_sp_reset`	If the value of Timer Action is Reset, this parameter tells the host the number of times the host resets the SP before it gives up	Positive integer up to 20	2
Admin State	`ilom_watchdog_admin_state`	The ILOM watchdog state	`enable|disable`	disable

Host Watchdog Parameters

The following host watchdog parameters can be set by the user in one of the following ways:

- Preferred method: Use the `ilomconfig` commands to change the parameters. See "Configure Host Watchdog Using `ilomconfig` Commands" on page 52. For more information on ilomconfig CLI commands, see "Host Watchdog ilomconfig Commands" in *Oracle Server CLI Tools User's Guide*.
- Manually edit the configuration file. See "Manually Edit the Configuration File" on page 52.

Property Name	Parameter in Configuration File	Description	Possible values	Default value
Reset Period	`host_watchdog_reset_period`	The interval at which the host informs Oracle ILOM that the host is alive (seconds). This value must be less than that of Timer Interval.	Positive integer that is less than the value of timer-value	60
Timer Action	`host_watchdog_timer_action`	The action Oracle ILOM takes when the host become unresponsive	`Warning|Reset|PowerOff|PowerCycle`	Warning
Timer Value	`host_watchdog_timer_value`	The interval at which Oracle ILOM performs the Timer Action if the host is unresponsive (seconds).	Positive integer, with a maximum of 6553	300
Admin State	`host_watchdog_admin_state`	The host watchdog state.	`enable|disable`	disable

Configuring the HMP Watchdog Agent

This section describes the HMP watchdog agent.

- "Commands to Control the HMP Watchdog Agent" on page 49
- "File Locations" on page 49

Commands to Control the HMP Watchdog Agent

The commands in the following table can be used to control the HMP watchdog agent.

Task	Command

Start the HMP watchdog agent.	`/etc/init.d/hmp_watchdogd start`
	or
	`service hmp_watchdog start`

Stop the HMP watchdog.	`/etc/init.d/hmp_watchdogd stop`
	or
	`service hmp_watchdog stop`

Stop and start the HMP watchdog agent.	`/etc/init.d/hmp_watchdogd restart`
	or
	`service hmp_watchdog restart`

Cause HMP watchdog agent to reread its configuration file.	`/etc/init.d/hmp_watchdogd reload`
	or
	`service hmp_watchdog reload`

Get the status of the watchdog agent.	`/etc/init.d/hmp_watchdogd status`
	or
	`service hmp_watchdog status`

File Locations

The following table lists the location of important HMP watchdog agent files.

File Type	Location
Configuration file	`/etc/`*ssm_directory*`/hmp_watchdogd.conf`

File Type	Location
	Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.
Log file	`/var/log/`*ssm_directory*`/hmp_watchdogd.log` Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.
PID file	`/var/run/sun-ssm-hmp-watchdogd.pid`
Binary file	`/usr/sbin/hmp_watchdogd`
ILOM watchdog state file	`/var/run/sun-ssm-ilom-watchdog-state`
Host watchdog state file	`/var/run/sun-ssm-host-watchdog-state`

Note - The ILOM and host watchdog state file locations are listed for your information only. These files are for internal use only.

▼ Change the Logging Level of the HMP Watchdog Agent

The logging level of the HMP watchdog agent can be changed by editing the agent's configuration file and changing the value of the `hmp_watchdogd_log_levels` property.

1. **Open the following file in a text editor:**

   `/etc/`*ssm_directory*`/hmp_watchdogd.conf`

   Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

2. **Change the value of the following parameter:**

   `hmp_watchdogd_log_levels=`*value*

   The following table shows the available log level values.

Value	Resulting Messages
CRIT	critical messages
ERROR	critical and error messages
WARNING	critical, error messages and warnings
NOTICE	critical, error messages, warnings, and notices

Value	Resulting Messages
INFO	critical, error messages, warnings, notices, and informational

3. **Save the configuration file.**

4. **Run the following command:**

```
/etc/init.d/hmp_watchdogd reload
```

Configure ILOM Watchdog Using `ilomconfig` Commands

This procedure provides basic instructions for using ilomconfig commands to configure the watchdog timer. For more details, see *Oracle Server CLI Tools User's Guide*.

Before You Begin You must be able to run commands on the host in administrator mode.

- **To modify the ILOM watchdog timer, use the following commands:**

  - **To enable or disable the ILOM watchdog:**

    ```
    ilomconfig enable|disable ilomwatchdog
    ```

  - **To modify the ILOM watchdog settings:**

  - **`ilomconfig modify ilomwatchdog --`***option*

    where *option* is one of:

Option	Possible values	Description
`--timer-action=`*action*	`Warning|Reset`	This sets the `ilom_watchdog_timer_action` parameter, which determines the action the ILOM watchdog takes when Oracle ILOM is not responsive.
`--number-sp-reset=`*numspreset*	Positive integer up to 20	This sets the `ilom_watchdog_number_sp_reset` parameter, which determines the number of times the SP can be reset consecutively.
`--query-interval=`*queryinterval*	Positive integer up to 2147483647	This sets the `ilom_watchdog_query_time_interval` parameter, which determines the number of seconds between querying for Oracle ILOM's health.

Configure Host Watchdog Using `ilomconfig` Commands

This procedure provides basic instructions for using ilomconfig commands to configure the host watchdog. For more details, see XREF CLI GUIDE.

Before You Begin You must be able to run commands on the host in administrator mode.

- **To modify the host watchdog parameters, use the following commands:**

  - **To enable or disable the host watchdog:**
    `ilomconfig enable|disable hostwatchdog`

  - **To modify the host watchdog settings:**

  - **`ilomconfig modify hostwatchdog --`***option*
    where *option* is one of:

Option	Possible Values	Description
`--timer-action=`*action*	`Warning|Reset|PowerOff|PowerCycle`	This sets the `host_watchdog_timer_action` parameter, which determines the action the host watchdog takes when the timer value expires.
`--timer-value=`*timervalue*	Positive integer up to 6553	This sets the `host_watchdog_timer_value` parameter, which determines the number of seconds before the timer expires.
`--reset-period=`*queryinterval*	Positive integer that is less than the value of timer-value parameter	This sets the `host_watchdog_reset_period` parameter, which determines the number of seconds before resetting the timer value.

Manually Edit the Configuration File

The recommended way to change parameters in the configuration file for ILOM watchdog and host watchdog is to use the `ilomconfig` command with the `ilomwatchdog` or `hostwatchdog` targets. Use this procedure only if you are not able to use the `ilomconfig` command.

1. **Open the following file in a text editor:**

   `/etc/`*ssm_directory*`/hmp_watchdogd.conf`

   Where *ssm_directory* is either `sun-ssm` or `ssm`, depending on your version of Oracle Hardware Management Pack.

2. **Change the value of any of the following parameters:**

   - **For ILOM watchdog:**
     - `ilom_watchdog_admin_state=`*value*
     - `ilom_watchdog_number_sp_reset=`*value*
     - `ilom_watchdog_timer_action=`*value*
     - `ilom_watchdog_query_time_interval=`*value*
   - **For host watchdog:**
     - `host_watchdog_admin_state=`*value*
     - `host_watchdog_timer_action=`*value*
     - `host_watchdog_timer_value=`*value*
     - `host_watchdog_reset_period=`*value*

   See "ILOM Watchdog Parameters" on page 47 and "Host Watchdog Parameters" on page 48 for information on defaults and available values for each parameter.

3. **Save the configuration file.**

4. **Run the following command:**

   ```
   /etc/init.d/hmp_watchdogd reload
   ```

Using the `itpconfig` Tool

The itpconfig tool enables you to configure a trap proxy to send traps from Oracle Integrated Lights Out Manager (ILOM) over the Host-to-ILOM Interconnect and forward the traps from the host server to a configurable destination. `itpconfig` can also enable or disable the Host-to-ILOM Interconnect, which is available on the latest Oracle servers. The Host-to-ILOM Interconnect provides a high speed internal interconnection between your server's Oracle ILOM service processors and the host, and must be enabled for the trap forwarding to function.

From Hardware Management Pack 2.2.6 onwards `itpconfig` is also supported on Microsoft Windows Server based operating systems. See "Configuring Trap Forwarding on Windows Servers" on page 61 for additional configuration information.

This section includes the following topics:

`itpconfig` Command Usage

The `itpconfig` commands must be run in administrator mode. The command syntax for `itpconfig` is:

`itpconfig` <*subcommand*> <*type*> [*options*]

When a command fails, it returns one of several failure codes listed in "Error Codes" on page 57.

Subcommands

The available itpconfig subcommands are:

Subcommand	Description
list	Show Oracle ILOM trap proxy or Host-to-ILOM Interconnect settings.
modify	Modify Oracle ILOM trap proxy settings.
enable	Enable trap forwarding or Host-to-ILOM Interconnect.
disable	Disable trap forwarding or Host-to-ILOM Interconnect.

See also "CLI Tools Command Syntax and Conventions" in *Oracle Server CLI Tools User's Guide*.

Supported Types

Type	Description
interconnect	Modify Host-to-ILOM interconnect settings. Mandatory options for enable or modify include: --ipaddress=*ipaddress* ILOM interconnect IP address --hostipaddress=*ipaddress* Host interconnect IP address --netmask=*netmask* Host-to-ILOM interconnect netmask
trapforwarding	Modify ILOM to send SNMP traps for all faults. Mandatory options for enable or modify include: --ipaddress=*ipaddress* IP Address to forward fault traps to --port=*port* Port number to forward fault traps to --community=*community* SNMP V2c community to use when forwarding fault traps

Options

The following options are available to all CLI Tools commands including itpconfig:

Short Option	Long Option	Description
-h	--help	Displays help information.
-V	--version	Displays the tool version.

Short Option	Long Option	Description
-q	--quiet	Suppresses informational message output and returns only error codes.
-y	--yes	Execute command without prompting for confirmation.

Error Codes

itpconfig generates error codes in a similar way to the Oracle Server CLI Tools. See "CLI Tools Error Codes" in *Oracle Server CLI Tools User's Guide*.

In addition, itpconfig generates the following error codes:

Code Number	Error Description
81	Oracle ILOM SNMP timeout.
82	Oracle ILOM SNMP failure.

These errors can occur if there are issues communicating with the Oracle ILOM SNMP service when enabling the trap proxy.

itpconfig Usage Scenario

The high level steps for enabling fault forwarding are:

1. Install the Oracle Hardware Management Agents and SNMP Plugins packages.

   See *Oracle Hardware Management Pack Installation Guide*

   These packages contain all the necessary software for itpconfig.
2. Enable the Host-to-ILOM Interconnect, required for itpconfig to function.

   The Host-to-ILOM Interconnect can be configured during installation. Alternatively you can use itpconfig, see "How to Enable Host-to-ILOM Interconnect" on page 58.
3. Enable the ILOM trap proxy.

   See "How to Enable Trap Forwarding" on page 59

Note - itpconfig uses ILOM Notification Alert Rule 15 to set up the trap forwarding. If this alert rule is in use, itpconfig fails. See "itpconfig Troubleshooting" on page 65 for a work around.

---

4. Start or restart the SNMP service daemon on the server.

   Refer to your OS documentation.
5. Start a trap listener on the destination server configured to listen to traps from the port and community described in the `itpconfig` arguments.

   Any faults generated by the service processor should now generate an SNMP trap which are sent to the destination SNMP trap listener.

Host-to-ILOM Interconnect Configuration Commands

The following procedures are covered in this section:

▼ How to Enable Host-to-ILOM Interconnect

The Host-to-ILOM Interconnect can be enabled during the Hardware Management Pack installation. See"Enabling the Host-to-ILOM Interconnect" in *Oracle Hardware Management Pack Installation Guide* for details.

Alternatively, you can use `itpconfig` to enable this feature and manage its properties.

Note - It is recommended that you use this command without any arguments and let `itpconfig` choose the settings. You can override the defaults with different IP and netmask addresses, but this is for advanced users only.

- **Issue the following command:**

`itpconfig enable interconnect [--ipaddress=`*ipaddress***`] [--netmask=`***netmask***`] [--hostipaddress=`***hostipaddress***`]`**

Option	Description	Example
`--ipaddress`	Oracle ILOM IP address. This address must be in the format: 169.254.x.x	**`169.254.175.72`**
`--netmask`	Oracle ILOM netmask.	**`255.255.255.0`**

Option	Description	Example
`--hostipaddress`	Host IP address. This address must be in the format: 169.254.x.x	**`169.254.175.73`**

How to Disable Host-to-ILOM Interconnect

To disable the Host-to-ILOM Interconnect between the host and Oracle ILOM, use the `itpconfig disable interconnect` command.

- **Issue the following command:**

```
itpconfig disable interconnect
```

How to List the Host-to-ILOM Interconnect Settings

To list the Host-to-ILOM Interconnect state and IP settings on both the Oracle ILOM and host side of the interconnect, use `itpconfig list interconnect`.

- **Issue the following command:**

```
itpconfig list interconnect
```

`itpconfig` Trap Forwarding Commands

This section includes the following procedures:

- "How to Enable Trap Forwarding" on page 59
- "How to Disable Trap Forwarding" on page 60
- "How to Disable Trap Forwarding" on page 60

How to Enable Trap Forwarding

- **To enable trap forwarding, issue the following command:**

itpconfig enable trapforwarding --ipaddress=*ipaddress* **--port=***port* **--community=***community*

Note - If the trap forwarding is already enabled, use the `itpconfig modify trapforwarding` command instead.

Mandatory options for `itpconfig enable trapforwarding` are:

Option	Description
`--ipaddress`	Sets the destination IP address for the forwarded trap. This can be loopback (127.0.0.1) or any other valid IP address. This must correspond to the configuration of the SNMP listener.
`--port`	Sets the destination port for the forwarded trap. There is no default value, but 162 is a common port value. This must correspond to the configuration of the SNMP listener.
`--community`	Sets the destination SNMP V2c community for the forwarded trap. This value must correspond to the configuration of the SNMP listener.

Example:

```
itpconfig enable trapforwarding --ipaddress=127.0.0.1 --port=1234 --
community=test
```

How to Disable Trap Forwarding

- **To disable `itpconfig` trap forwarding, issue the following command:**

  itpconfig disable trapforwarding

  The disable command takes no additional parameters and disables the trap forwarding operation on both ILOM and the host.

How to List Trap Forwarding Settings

- **To list itpconfig trap forwarding settings, issue the following command:**

  itpconfig list trapforwarding

  This returns output similar to the following:

```
Trap Forwarding
================ Trap
Forwarding is enabled
```

```
Trap Forwarding Destination: 127.0.0.1
Trap Forwarding Port: 162
Trap Forwarding Community: test
```

The list command takes no additional parameters.

Configuring Trap Forwarding on Windows Servers

This procedure explains how to configure ILOM trap forwarding a server running a Windows based operating system using `itpconfig`. This process requires configuring the server which sends the traps, referred to as the source server in this procedure, and the server which receives the trap, referred to as the destination in this procedure.

▼ How to configure trap forwarding on Windows servers

1. **Log in to the source server. You must have Administrator privileges.**

2. **Use the `itpconfig.exe enable trapforwarding` subcommand to enable the trap proxy.**

   `itpconfig.exe enable trapforwarding --ipaddress=`*destination* **`--port=162 --community=`***trap_community*

   where *destination* is the IP address of the server that should receive the traps, and *trap_community* is the SNMP trap community that the destination is listening for.

   Note - the port number of 162 can not be modified on Windows.

3. **If either of the source or destination servers use a firewall, configure the firewall rules on both to allow incoming traps.**

   a. **Go to Control panel and select Firewall.**

   b. **Click on Advanced Setting and then Inbound Rules on the left panel. The rules are shown in the right panel.**

   c. **Enable Inbound Rules for both private and domain by right clicking SNMP Trap Service and selecting Enable.**

4. **Restart the SNMP Trap service and the Oracle Hardware Management Agent service.**

    a. **Go to Server Manager, select Services.**

    b. **Find the SNMP Trap service and start/restart it.**

    c. **Find the Oracle Server Hardware Management Agent service and start/restart it.**

Using Oracle Hardware Management Pack to Monitor Disk Diagnostic Events

This section describes enhanced diagnostic features added to Oracle Hardware Management Pack to collect disk error and SMART events from disks attached to the Sun Storage 6 Gb SAS PCIe HBA, Internal (SGX-SAS6-INT-Z) and store them in the hardware management agent event log.

Monitoring Disk Events

As of Oracle Hardware Management Pack 2.3.2.2, enhanced diagnostic features have been added to collect disk error and SMART events from disks attached to the Sun Storage 6 Gb SAS PCIe HBA, Internal (SGX-SAS6-INT-Z), whether independent or in a RAID volume.

These enhanced diagnostic events are captured and logged in `/var/log/ssm/event.log` when the hardware management agent is running.

The following table lists the enhanced diagnostic events being logged.

Event Name in Log	Description
PD_RECOVERED_ERROR	A disk recovered error was detected.
PD_BAD_DEVICE_FAULT	A non-recoverable drive failure was detected by the device while performing a command.
PD_MEDIA_ERROR	A medium error was detected by the device that was non-recoverable.
PD_DEVICE_ERROR	A non-recoverable hardware failure was detected by the device. The device may be offlined or degraded.
PD_TRANSPORT_ERROR	A path to the device has been unconfigured due to transport instability.

Event Name in Log	Description
PD_OVER_TEMPERATURE	Disk SMART process reports a critical temperature.
PD_SELF_TEST_FAILURE	One or more disk SMART self tests failed.
PD_PREDICTIVE_FAILURE	SMART health-monitoring firmware reported that a disk failure is imminent.

The controller polls each physical disk at regular intervals. If a disk has encountered an error, an event is generated by the controller. The hardware management agent captures that event and enters it in the hardware management event log.

To view the event information in the hardware management event log, type:

```
# view /var/log/ssm/event.log
```

For enhance diagnostic disk events, you will see information similar to:

```
Thu Apr 30 12:32:31 2015:(CLI) Event Name  : PD_MEDIA_ERROR
Thu Apr 30 12:32:31 2015:(CLI) Event Description : A medium error was
detected by the device that was non-recoverable.
Thu Apr 30 12:32:31 2015:(CLI) ASC  : 0x10
Thu Apr 30 12:32:31 2015:(CLI) ASCQ : 0x3
Thu Apr 30 12:32:31 2015:(CLI) Sense Key : 0x3
Thu Apr 30 12:32:31 2015:(CLI) Source : LSI
Thu Apr 30 12:32:31 2015:(CLI) SAS Address : 0x5000cca01200fadd
Thu Apr 30 12:32:31 2015:(CLI) LSI Description : Unexpected sense: PD
0c(e0xfc/s1) Path 5000cca01200fadd, CDB: 2f 00 00 fc 4d 42 00 10 00 00,
Sense: 3/10/03
Thu Apr 30 12:32:31 2015:(CLI) Event TimeStamp : 04/30/2015 ; 19:30:25
Thu Apr 30 12:32:31 2015:(CLI) Node ID : 00000000:12
Thu Apr 30 12:32:31 2015:(CLI) Nac Name : /SYS/HDD1
Thu Apr 30 12:32:31 2015:(CLI) Serial Number : 001015N0JPXA   PMG0JPXA
Thu Apr 30 12:32:31 2015:(CLI) WWN No : PDS:5000cca01200fadd
Thu Apr 30 12:32:31 2015:(CLI) Disk Model : H106030SDSUN300G
```

You can then use the information in the event listing to determine which physical disk in the system has the issue. Information such as the Oracle ILOM Nac Name (which matches the label on the front panel of the system) and drive Serial Number help you identify the disk and its drive slot in the system.

Note - For PD_OVER_TEMPERATURE, PD_SELF_TEST_FAILURE and PD_PREDICTIVE_FAILURE events, use Oracle ILOM to configure proactive alerts.

For the other disk diagnostic events described in this document, it is up to the administrator to check the hardware management event log for these disk events when a disk problem is suspected. There is currently no alert mechanism to proactively announce these events.

Troubleshooting Management Agents

This section provides tips and solutions for the most common problems you might encounter when working with Management Agents. The section contains:

General Management Agents Troubleshooting

The best way to troubleshoot problems with Management Agents is to review the log files.

The Hardware Management Agent stores log information in the `hwmgmtd.log` file.

For more information on the `hwmgmtd.log` file, see "Configure the Hardware Management Agent Logging Level" on page 19.

`itpconfig` Troubleshooting

`itpconfig` uses ILOM Notification Alert Rule 15 to set up the trap forwarding. If this alert rule is in use, `itpconfig` fails with error code 83 . This error is caused when you try to run `itpconfig` when ILOM Notification Alert Rule 15 is already defined on the system.

To work around this, set the destination IP address of ILOM Notification Alert Rule 15 to 0.0.0.0.

Oracle Solaris Operating System Troubleshooting

The following topics can help you to identify and solve problems when using the Hardware Management Pack on Oracle Solaris OS.

This section covers the following topics:

Issues Installing with pkgadd

When using pkgadd(1M) during installation, if you encounter the following error message:

```
#Waiting for up to <300> seconds for package administration commands to become
available (another user is administering packages on zone <XXX>)
```

An interruption of the pkgadd(1M) process can leave an outstanding packaging lock file, which blocks further use of the pkgadd (1M) command. Before attempting another installation, remove the packaging lock file.

▼ How to Remove a Packaging Lock File

1. **At the command prompt, type the following:**

   ```
   svccfg list
   ```

   If you see/TEMP/application/management/hwmgmtd listed in the output then delete the file by typing the following:

   ```
   svccfg delete TEMP/application/management/hwmgmtd
   ```

2. **Type the following:**

   ```
   svccfg list
   ```

   You should no longer see TEMP/application/management/hwmgmtd listed.

3. **Remove the packages by typing the following:**

   ```
   pkgrm SUNWssm-hwmgmt-config
   ```

   You should now be able to install SUNWssm-hwmgmt-config.

Linux Troubleshooting

The following topics can help you to identify and solve problems when using the Hardware Management Pack on Linux.

This section covers the following topics:

Hardware Management Agent Service Fails to Start

After installing the Hardware Management Agent on SUSE Linux Enterprise, you might encounter the following:

```
Starting Sun HW agent services: . . . . . failed
```

In addition, there might be a line in the Hardware Management Agent log file similar to the following:

```
(hwagentd_poller.c:334:hwagent_bmc_response_test):Unable to reach the KCS
interface over ipmitool-hwagentd.
```

This problem occurs when the IPMI device drivers are not installed. Hardware Management Agent uses the IPMI drivers to access the KCS interface.

▼ How to Solve Issues With IPMI Device Drivers

1. **Install an IPMI system such as OpenIPMI which provides device drivers for full access to IPMI information.**
2. **Start the Hardware Management Agent.**

Hardware Management Agent Service Status Dead

After installing the Hardware Management Agent on Red Hat Enterprise Linux, the hwmgmtd service starts but you see something similar to the following:

```
/etc/init.d/hwmgmtd start

Starting Sun HW agent services: . . . . . . . . [ OK ]

/etc/init.d/hwmgmtd status

hwmgmtd dead but subsys locked
```

In addition, there may be a line in the Hardware Management Agent similar to the following:

```
hwagentd_poller.c:334:hwmgmtd_bmc_response_test):Unable to reach the KCS
interface over ipmitool-hwmgmtd.
```

This problem occurs when the IPMI device drivers have not been installed. Hardware Management Agent uses the IPMI drivers to access the KCS interface.

Solution: Install an IPMI system such as OpenIPMI, which provides device drivers for full access to IPMI information.

How to Solve Issues with IPMI Device Drivers

1. **Install an IPMI system such as OpenIPMI which provides device drivers for full access to IPMI information.**
2. **Start the Hardware Management Agent.**

Index

M

O

S

T

W